



# Apache+PHP5+Mysql

## ::[Instalação]::

por : **Fred Cox Junior** <fred.cox@terra.com.br>  
data : 26/08/2004  
email: **fred.cox@terra.com.br**

## 1. INTRODUÇÃO

**E**ste documento tem por objetivo detalhar passo a passo o processo de instalação dos seguintes programas em ambiente GNU/Linux:

- Apache versão - 1.3.31
- PHP - 5.0.1
- MySQL - 4.0.20
- Bluefish - 0.13

Primeiro veremos a compilação e instalação dos programas para depois prepararmos o ambiente para programação em PHP com banco de dados MySQL.

Utilizei o **Slackware 10**, distro que uso desde 1998 quando tive os primeiros contatos com a versão 3.5. Procurei abordar desde a compilação até a configuração e execução dos programas e processos necessários para o funcionamento da tríade **apache+php+mysql**, de uma forma geral, procurei compatibilizar o processo de instalação para o padrão **GNU/Linux**. Decidi não utilizar programas pré-compilados, nem receitas de bolo

específicas ou restritas para determinadas distros. Procurei abranger e detalhar o máximo possível, com as distribuições Linux, mais utilizadas atualmente. A instalação será realizada pelo método tradicional a partir dos códigos fontes dos respectivos programas, compilando e instalando os aplicativos com os comandos ; **./configure <opções>; make; make install.**

Todos os procedimentos para compilação e instalação são referenciados no diretório **/usr/src**, de forma que, você precisará de privilégios de root para instalar e dependendo das permissões, até mesmo copiar os sources para este diretório.

## **2. Instalação do MySQL**

Quando estava escrevendo este artigo a última versão estável do **MySQL** é a **4.0.20**, portanto, iremos utilizar esta versão como referência nos procedimentos de instalação. É importante que o leitor siga na íntegra os passos para compilação e instalação e só depois faça a configuração e preparação do ambiente, haja vista que cada tópico aqui abordado é dependente do anterior. Para realizar a integração do PHP com o MySQL, é necessário antes de instalar o PHP ter o MySQL previamente instalado e configurado, e, para instalar o Apache com suporte a PHP, é necessário ter os Sources do PHP para que este gere os módulos necessários para o funcionamento, portanto a instalação do servidor de banco de dados é necessariamente primordial antes mesmo de começar a instalar a dupla Apache/PHP.

Baixe a última versão dos sources do **MySQL** no seguinte site:

**<http://www.mysql.com>**

No shell, digite:

```
[/usr/src]$ wget \
http://www.linorg.usp.br/mysql/Downloads/MySQL- 4.0/mysql-
4.0.20.tar.gz
```

Descompacte, compile e acesse o diretório com os seguintes comandos:

```
[/usr/src]$ tar -zxvf mysql- 4.0.20.tar.gz
[/usr/src]$ cd mysql- 4.0.20/
[/usr/src/mysql- 4.0.20]$ ./configure --prefix=/usr/local/mysql
[/usr/src/mysql- 4.0.20]$ make
[/usr/src/mysql- 4.0.20]$ make install
[/usr/src/mysql- 4.0.20]$ ldconfig
[/usr/src/mysql- 4.0.20]$ cp support-files/my-medium.cnf /etc/my.cnf
```

Agora que já compilamos e instalamos o mysql, você deverá criar um usuário com poucos privilégios para executar o daemon do MySQL e em seguida setar as permissões nos diretórios para usuário e grupo mysql.

No shell, digite:

```
[/usr/src/mysql- 4.0.20]$ cd /usr/local/mysql/
[/usr/local/mysql]$ groupadd mysql
[/usr/local/mysql]$ useradd -g mysql mysql
[/usr/local/mysql]$ chown -R root .
[/usr/local/mysql]$ chown -R mysql var
[/usr/local/mysql]$ chgrp -R mysql .
```

Execute o script **mysql\_install\_db** que criará as tabelas iniciais do servidor de banco de dados.

```
[/usr/local/mysql]$ $ bin/mysql_install_db --user=mysql
```

O comando acima criará as tabelas iniciais e de **samples** do seu servidor de banco de dados. O script que deve ser executado para subir o daemon (processo) do banco de dados é o

**mysql.server** que está localizado em:

*/usr/src/mysql- 4.0.20/support- files/*

Se o leitor quiser que o MySQL inicie automaticamente no boot, copie o script **mysql.server** que está localizado no diretório acima, renomeando-o para **/etc/rc.d/rc.mysql** ou **/etc/init.d/rc.mysql** (isto vai depender da sua distribuição Linux), ou então fazer referência a ele em **rc.local**, por exemplo:

```
[/usr/src/mysql- 4.0.20/support- files]$ cp mysql.server /etc/rc.d/rc.mysql
[/usr/src/mysql- 4.0.20]$ chmod +x /etc/rc.d/rc.mysql
```

Por fim, você precisa definir a senha do usuário root de seu servidor de banco de dados.

Primeiramente, inicie o servidor de banco de dados, digite no shell:

```
[/etc/rc.d ]$ ./rc.mysql start
```

Se tudo der certo, o servidor estará na escuta na porta **3306**, passe um escaner de portas **nmap**, que pode ser obtido no endereço **<http://www.insecure.org/nmap/>**, no seu computador; veja o exemplo:

```
[/etc/rc.d ]$ nmap -sT -O localhost -p 3306
Starting nmap 3.50 ( http://www.insecure.org/nmap/ ) at 2004-08-16
20:32 BRT
Warning: OS detection will be MUCH less reliable because we did not find at
least 1 open and 1 closed TCP port
Interesting ports on localhost (127.0.0.1):
PORT      STATE SERVICE
3306/tcp  open  mysql
Device type: general purpose
Running: Linux 2.4.X/2.5.X
OS details: Linux Kernel 2.4.0 - 2.5.20
Uptime 0.081 days (since Mon Aug 16 18:36:38 2004)
```

Quando você se conecta a um MySQL Server, normalmente é solicitada uma senha de usuário. Esta informação poderá ser lida no momento em que ela trafegar na rede do cliente para o servidor. Para evitar os **"scripts-kiddies"** que ficam testando ferramentas e exploits no seu servidor, você deve filtrar a porta **3306** permitindo conexão apenas de hosts confiáveis (Trusted hosts), via firewall iptables (editando a chain INPUT), na qual foge ao escopo do artigo, ou então desabilitar a escuta da porta **3306**, descomentando (retirando o sinal #) a linha referente a **skip-networking** no arquivo **/etc/my.cnf**.

Quando desabilitamos a escuta da porta **3306**, somente serão permitidas consultas a partir de do seu servidor (localhost). Ninguém, nem mesmo o root poderá conectar remotamente ao Servidor MySQL, porém você poderá utilizar o **PHPMysqlAdmin**, que é um excelente front-end do mysql, para gerenciamento de tabelas e administração do servidor do banco de dados.

Abra o arquivo **/etc/my.cnf** com seu editor preferido (vi, pico, joe, etc...) e procure pela seção seguinte:

```
# Don't listen on a TCP/IP port at all. This can be a security enhancement,  
# if all processes that need to connect to mysqld run on the same host.  
# All interaction with mysqld must be made via Unix sockets or named pipes.  
# Note that using this option without enabling named pipes on Windows  
# (via the "enable-named-pipe" option) will render mysqld useless!  
#  
#skip-networking
```

Descomente a linha **skip-networking** e reinicialize o servidor MySQL.

```
# Don't listen on a TCP/IP port at all. This can be a security enhancement,  
# if all processes that need to connect to mysqld run on the same host.  
# All interaction with mysqld must be made via Unix sockets or named pipes.  
# Note that using this option without enabling named pipes on Windows  
# (via the "enable-named-pipe" option) will render mysqld useless!  
#  
skip-networking
```

O MySQL criptografa as senhas dos usuários através do algoritmo hash md5 semelhante ao processo de armazenamento de senhas do Linux.

Após a instalação do MySQL, por padrão, há dois usuários; um superusuário (root) e outro usuário padrão. Você deverá excluir o usuário padrão e definir uma senha para o root.

Para defini a senha do root (superusuário) e apagar o usuário padrão, siga os passos a seguir. Utilizaremos o mysql monitor, que é o front-end em modo texto, que vem disponível na instalação do MySQL.

No shell digite os seguintes comandos:

- ps: observe que *mysql>* corresponde ao prompt do MySQL monitor.

```
[/usr/local/mysql/bin ]$ /usr/local/mysql/bin/mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 1 to server version: 4.0.20-log

Type 'help;' or '\h' for help. Type '\c' to clear the buffer.

mysql> use mysql
Database changed
mysql> delete from user where user='';
Query OK, 2 rows affected (0.09 sec)
mysql> update user set password=password('thc691') where user='root';
Query OK, 1 row affected (0.03 sec)
Rows matched: 2  Changed: 1  Warnings: 0
mysql> flush privileges;
Query OK, 0 rows affected (0.00 sec)
mysql> quit
```

### 3. Instalação do Apache 1.3.31

O apache é o servidor de web mais utilizado no mundo. No momento que escrevia este artigo, a Netcraft - <http://www.netcraft.com> -, órgão que registra e publica estatísticas de utilização de sistema na Internet, registrava que **67.70%** dos servidores no mundo rodavam o Apache web server. Se você acompanhar a curva de adoção do apache e da utilização do linux, verá que são praticamente idênticas. O sucesso do apache se deve a sua simplicidade, robustez, segurança e alto desempenho.

A escolha da versão **1.3.31** se deve à segurança e estabilidade. Apesar da versão **2.x** rodar sem nenhum problema, a grande maioria não vê motivo para trocar de versão.

A instalação do apache que iremos abordar será do tipo standalone, haja vista, que este tipo de instalação/configuração possibilita um melhor balanceamento das conexões aumentando assim a performance. Os requisitos necessários de hardware para utilizar o apache são um computador pentium 233 com 48 Mbytes de memória RAM, embora o desenvolvedor afirme que a configuração mínima é um 486 com 8 Mb de RAM; é até possível rodar o apache com esta configuração, desde que a maioria dos recursos estejam desabilitados.

Iremos configurar o apache com as seguintes características:

- a) Suporte a CGI's e Perl***
- b) Suporte páginas dinâmicas PHP***
- c) Autenticação de usuários (.htaccess)***
- d) hospedar banco de dados MySQL***

obs: A grande maioria dos procedimentos seguintes requer o usuário root.



Baixe a versão 1.3.31 do apache em <http://www.apache.org> ou puxe direto deste endereço:

[http://ftp.pucpr.br/apache/httpd/apache\\_1.3.31.tar.gz](http://ftp.pucpr.br/apache/httpd/apache_1.3.31.tar.gz)

Descompacte o arquivo `apache_1.3.31.tar.gz` e entre no diretório:

```
[/usr/src] $ tar -zxvf apache_1.3.31.tar.gz  
[/usr/src]$ cd apache_1.3.31/
```

Vamos escolher a seguinte estrutura para o apache:

<code>/var/www/cgi-bin</code>	<i>Diretório de execução de scripts cgi;</i>
<code>/var/www/htdocs</code>	<i>Será o diretório onde ficarão as páginas html e os scripts php;</i>
<code>/var/log/apache</code>	<i>diretório de logs de acesso do apache;</i>
<code>/etc/apache</code>	<i>Aqui ficará os arquivos de configuração do servidor de web (httpd.conf, srm.conf e access.conf);</i>
<code>/usr/local/apache</code>	<i>será o local onde ficará os programas e arquivos necessários para inicialização do daemon httpd.</i>
<code>UID/GUID</code>	<i>nobody será o usuário que executará o daemon httpd</i>

Conforme configuração escolhida na tabela acima, execute os procedimentos no diretório dos sources do apache:

```
[/usr/src/apache_1.3.31/]$ ./configure \  
- - prefix=/usr/local/apache - - htdocsdir=/var/www/htdocs \  
- - cgidir=/var/www/cgi- bin - - logfiledir=/var/log/apache \  
- - sysconfdir=/etc/apache - - server- uid=nobody - - server- gid=nobody \  
- - enable- module=so  
[/usr/src/apache_1.3.31/]$ make  
[/usr/src/apache_1.3.31/]$ su  
[/usr/src/apache_1.3.31/]$ make install
```

Após a compilação e instalação, verifique se o seu sistema

possui o usuário e grupo **nobody** com os seguintes comandos:

```
$ grep 'nobody' /etc/passwd  
$ grep 'nobody' /etc/group
```

A conta **nobody** corresponde a um usuário genérico que não possui direitos no sistema, apenas privilégio no daemon do Apache.

Se precisar adicionar o usuário e grupo nobody, execute:

```
$ groupadd nobody  
$ useradd -g nobody -d / nobody
```

Para testar a instalação default do apache:

No shell digite:

```
[/usr/local/apache/bin]$ ./httpd - V
Server version: Apache/1.3.31 (Unix)
Server built: Aug 21 2004 11:12:00
Server's Module Magic Number: 1XXXXXXX
Server compiled with....
- D HAVE_MMAP
- D HAVE_SHMGET
- D USE_SHMGET_SCOREBOARD
- D USE_MMAP_FILES
- D HAVE_FCNTL_SERIALIZED_ACCEPT
- D HAVE_SYSVSEM_SERIALIZED_ACCEPT
- D SINGLE_LISTEN_UNSERIALIZED_ACCEPT
- D DYNAMIC_MODULE_LIMIT=64
- D HARD_SERVER_LIMIT=256
- D HTTPD_ROOT="/usr/local/apache"
- D SUEXEC_BIN="/usr/local/apache/bin/suexec"
- D DEFAULT_PIDLOG="logs/httpd.pid"
- D DEFAULT_SCOREBOARD="logs/httpd.scoreboard"
- D DEFAULT_LOCKFILE="logs/httpd.lock"
- D DEFAULT_ERRORLOG="logs/error_log"
- D TYPES_CONFIG_FILE="/etc/apache/mime.types"
- D SERVER_CONFIG_FILE="/etc/apache/httpd.conf"
- D ACCESS_CONFIG_FILE="/etc/apache/access.conf"
- D RESOURCE_CONFIG_FILE="/etc/apache/srm.conf"
```

Se tudo correu bem você verá a versão instalada do apache e a data e hora de compilação e as opções escolhidas no configure.

**ps:** Não é objetivo deste artigo é explicar sobre as possíveis configurações do servidor de web apache, nem de abranger todas as opções de customizações deste servidor. Se o leitor quiser obter mais informações, visite o site <http://www.apache.org> para maiores detalhes.

## 4. Instalação do PHP5

Inicialmente o leitor deve copiar a última versão do código fonte do PHP no seguinte site: <http://www.php.net/>

Algumas das novas funcionalidades adicionadas no PHP5 estão relacionadas na tabela abaixo:

- Zend Engine 2
- Suporte a classes abstratas e interfaces
- PHP5 foi totalmente reescrito com libxml2
- Suporte a exceptions e exceptions handling

No momento em que escrevia este livro a última versão do PHP foi a **5.0.1**.

Baixe o código fonte da versão mais recente do PHP com o wget:

No shell digite:

```
[/usr/src]$ wget http://br2.php.net/get/php-5.0.1.tar.gz/from/br.php.net/mirror
```

Descompacte os fontes do PHP no mesmo diretório que você copiou o `apache_1.3.31.tar.gz`, no nosso caso: `/usr/src`

No shell digite:

```
[/usr/src]$ tar -zxvf php-5.0.1.tar.gz  
[/usr/src]$ cd php-5.0.1/
```

Escolhi por motivos óbvios instalar o PHP com suporte a banco de dados **MySQL** e **Apache**. Para compilar e instalar o PHP5 siga os passos seguintes:

```
[/usr/src/php-5.0.1]$ ./configure --with-mysql=/usr/local/mysql \  
--with-apxs=/usr/local/apache/bin/apxs  
[/usr/src/php-5.0.1]$ make  
[/usr/src/php-5.0.1]$ su  
[/usr/src/php-5.0.1]$ make install
```

Se tudo deu certo, o leitor deve copiar o arquivo **php.ini.dist** para **/etc/apache/php.ini**. Este arquivo de configuração define as diretrizes de funcionamento e configuração do PHP5. Também será necessário realizar três alterações no arquivo de configuração do **apache** (**httpd.conf**), acrescentar suporte a **PHP5**, modificar os parâmetros **DirectoryIndex** adicionar a extensão (DSO) do **respectivo módulo**. Siga os passos seguintes

```
[/usr/src/php- 5.0.1/]$ cp php.ini-dist /etc/apache/php.ini
```

Procure pela seção abaixo no seu **/etc/apache/httpd.conf**:

```
# Dynamic Shared Object (DSO) Support
#
# To be able to use the functionality of a module which was built as
# a DSO you
# have to place corresponding 'LoadModule' lines at this location so
# the
# directives contained in it are actually available _before_ they are
# used.
# Please read the file http://httpd.apache.org/docs/dso.html for more
# details about the DSO mechanism and run 'httpd -l' for the list of
# already
# built-in (statically linked and thus always available) modules in
# your httpd
# binary.
#
# Note: The order in which modules are loaded is important.  Don't
# change
# the order below without expert advice.
#
# Example:
# LoadModule foo_module libexec/mod_foo.so
```

Acrescente a seguinte linha:

```
LoadModule php5_module          libexec/libphp5.so
```

Deve ficar assim:

```

# Dynamic Shared Object (DSO) Support
#
# To be able to use the functionality of a module which was built as
a DSO you
# have to place corresponding 'LoadModule' lines at this location so
the
# directives contained in it are actually available before they are
used.
# Please read the file http://httpd.apache.org/docs/dso.html for more
# details about the DSO mechanism and run 'httpd -l' for the list of
already
# built-in (statically linked and thus always available) modules in
your httpd
# binary.
#
# Note: The order in which modules are loaded is important. Don't
change
# the order below without expert advice.
#
# Example:
# LoadModule foo_module libexec/mod_foo.so
# Linkando com a biblioteca
LoadModule php5_module          libexec/libphp5.so

```

Você precisa informar também ao servidor quais são os arquivos ou scripts que ele irá invocar o interpretador PHP5, para isto, adicione as seguintes linhas no final do seu **/etc/apache/httpd.conf**.

```

#Definindo tipos de arquivos php
AddType application/x-httpd-php .php .phtml
AddType application/x-httpd-php-source .phps
# Feito

```

Agora defina também a seção **DirectoryIndex**, que são os arquivos que o apache irá procurar por default num diretório de web.

```
#
# DirectoryIndex: Name of the file or files to use as a pre-written
HTML
# directory index.  Separate multiple entries with spaces.
#
<IfModule mod_dir.c>
    DirectoryIndex index.html
</IfModule>
```

Acrescente o seguinte:

```
#
# DirectoryIndex: Name of the file or files to use as a pre-written
HTML
# directory index.  Separate multiple entries with spaces.
#
<IfModule mod_dir.c>
    DirectoryIndex index.html index.php index.phtml
</IfModule>
```

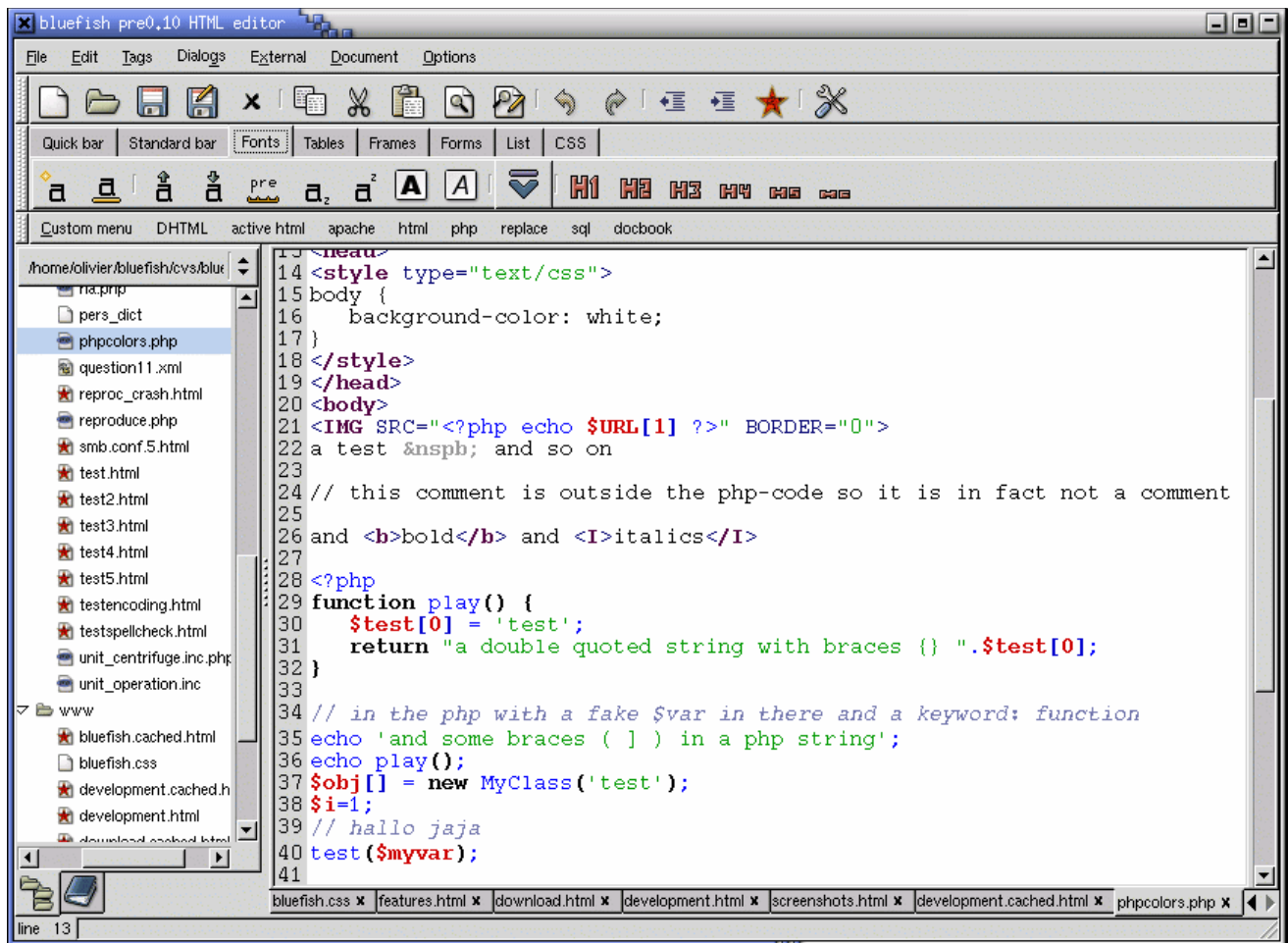
Caso o leitor queira que o servidor de web inicie no boot, crie um link simbólico rc.httpd no diretório /etc/rc.d ou /etc/init.d; isto irá depender exclusivamente de sua distribuição.

## 5. Instalação do Bluefish 0.13

Bluefish é um poderoso IDE editor html/php, distribuído sob licença GNU/GPL, com as seguintes características:

- Capacidade de abrir múltiplos documentos
- Suporte à arquivos remotos utilizando gnome-vfs
- Syntax highlighting
- Geração de Thumbnails para página de screenshots ou alguns fotográficos.

- Assistentes de utilização, etc...



O editor pode ser obtido gratuitamente através do seguinte endereço:

**<http://bluefish.openoffice.nl>**

A instalação do Bluefish é simples e não requer parâmetros adicionais. Apenas descompacte num diretório de fontes, compile e instale o aplicativo com os seguintes comandos:



```
[/usr/src/]$ tar -zxvf bluefish- 0.13.tar.gz
[/usr/src/]$ cd bluefish- 0.13/
[/usr/src/bluefish- 0.13]$ ./configure
[/usr/src/bluefish- 0.13]$ make
[/usr/src/bluefish- 0.13]$ su
[/usr/src/bluefish- 0.13]$ make install
```

Para iniciar o aplicativo, digite **bluefish** na linha de comando.

## 6 Testando a instalação

Em primeiro lugar iremos verificar se ocorreu algum erro de sintaxe na edição do arquivo **/etc/apache/httpd.conf**. Para efetuar esta verificação, execute o comando abaixo:

```
[/usr/]$ /usr/local/apache/bin/apachectl configtest
Syntax OK
```

Se a saída do comando **NÃO** foi Syntax OK, volte para a seção anterior, corrija o erro de sintaxe e verifique novamente com o comando acima.

Inicie o servidor de web digitando o seguinte:

```
[/usr/]$ /usr/local/apache/bin/apachectl start
/usr/local/apache/bin/apachectl start: httpd started
```

Rode um scanner de portas, nmap conforme visto na seção 6.1, no seu computador ou dê um **netstat** para certificar de que o apache está escutando na porta 80/tcp.

```
[/usr/]$ netstat -an|grep 0.0.0.0:80
tcp      0      0 0.0.0.0:80          0.0.0.0:*          LISTEN
```

Observe que a saída do comando informa que há um servidor aceitando conexões na porta 80.

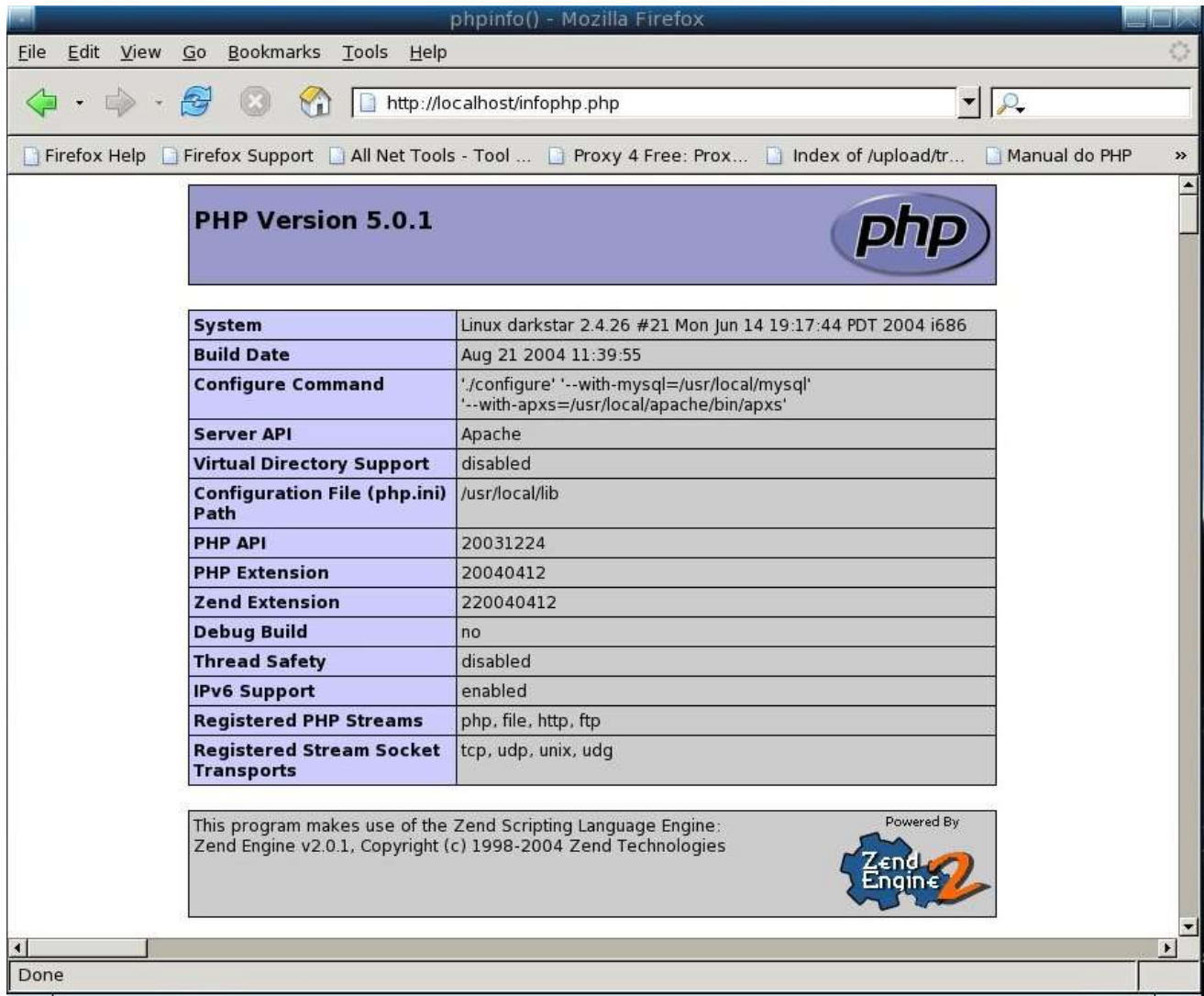
Inicie seu browser preferido, Netscape, Mozilla ou firefox, e digite o seguinte endereço: **http://localhost**; o navegador deverá exibir uma página semelhante a figura abaixo, que é página padrão da instalação do apache e está localizada no diretório raiz do seu servidor de web, no nosso caso, **/var/www/htdocs**.



Para saber se o php está realmente funcionando, crie um script com o seguinte conteúdo:

```
1 <?php
2     phpinfo();
3 ?>
```

Salve o arquivo em `/var/www/htdocs` com o seguinte nome: **infophp.php**. Inicie o browser e aponte para o seguinte endereço: **`http://localhost/infophp.php`**. O navegador deverá exibir uma página com o seguinte conteúdo:



Dúvidas, críticas e sugestões serão bem vindas.

**Fred Cox Junior**

**`fred.cox@terra.com.br`**

**`http://paginas.terra.com.br/lazer/fredcox`**